



Progetto:

EvenTrento

Titolo del documento:

Implementazione

Autore:

Togni Roberto

Document Info:

Doc. Name	D3-EvenTrentoImplementazione	Doc. Number	D3 V0.1
Description	Documento di descizione dell'implementazione		

Dipartimento di Ingegneria e Scienza dell'Informazione

Contents

1	Scopo del Documento	i
2	User Stories	ii
3	User Flow	iii
4	Web APIs	vi
5	Implementation	vii
5.1	Repository Implementation	vii
5.2	Branching strategy e organizzazione del lavoro	vii
5.3	Dependencies	vii
5.4	Database	viii
5.5	Testing	i
6	Front-End	ix
7	Deployment	ix

1 Scopo del Documento

Il presente documento ha lo scopo di riportare le informazioni riguardanti lo sviluppo di una parte dell'applicazione EvenTrento. Partendo dalle user stories, il documento presenta un walk-throgh delle funzionalità effettivamente implementate, con un focus sull'interazione dell'utente con queste ultime. Prosegue dunque con la descrizione delle web APIs dell'applicazione, nonché della modalità di sviluppo collaborativo, del database e del testing delle varie funzionalità. Infine, viene presentato il front-end realizzato, accompagnato da una descrizione della modalità di deployment utilizzata.

2 User Stories

Le seguenti tabelle raccolgono una serie di User Stories divise in diverse epiche. Ad ogni user story è associata una priorità (ad un numero più piccolo corrisponde una priorità maggiore) rappresentante l'importanza della funzionalità in questione per l'operatività del sistema.

Profilo			
Id	Nome	User Story	Priorità
A1	Registrazione	Come utente, voglio registrarmi per utilizzare il sistema	1
A2	Login	In qualità di utente, devo essere in grado di effettuare il login	1
A3	Visualizzazione profilo	Come utente, devo essere in grado di visualizzare lo storico eventi, le mie informazioni personali, e gli eventi salvati	3

Table 1: User stories relative all'epica "profilo".

Eventi			
Id	Nome	User Story	Priorità
B1	Lista eventi	In qualità di utente, devo essere in grado di scorrere una lista di eventi	2
B2	Filtri eventi	In qualità di utente, devo essere in grado di filtrare gli eventi in base alla data	2
B3	Creazione evento	In qualità di organizzatore, devo essere in grado di creare un nuovo evento in un determinato luogo	2
B4	Condivisione evento	In qualità di utente, devo essere in grado di condividere un evento tramite link	4
B5	Pagamento	In qualità di utente, devo essere in grado di effettuare il pagamento per un evento a cui voglio iscrivermi	2
B6	Aggiornamento evento	In qualità di organizzatore, devo essere in grado di aggiornare la descrizione di un evento	3
B7	Statistiche evento	In qualità di organizzatore, devo essere in grado di visualizzare gli iscritti all'evento	4
B8	Salvataggio evento	In qualità di utente, devo essere in grado di salvare un evento in modo da poterlo visualizzare nella mia area personale	3

Table 2: User stories relative all'epica "eventi".

Spazi			
Id	Nome	User Story	Priorità
C1	Aggiunta spazio	In qualità di owner, devo essere in grado di pubblicare uno spazio disponibile	2
C2	Rimozione spazio	In qualità di owner, devo essere in grado di rimuovere uno spazio esistente	2
C3	Visualizzazione spazi	In qualità di organizzatore, devo essere in grado di visualizzare gli spazi disponibili	3

Table 3: User stories relative all'epica "spazi".

4 Web APIs

5 Implementation

5.1 Repository Implementation

Il codice del progetto, disponibile su GitHub, è organizzato secondo questo schema:

```
.
D1
D2
D3
D4
front_end
  index.html
  pages
node_app
  app.js
  authentication.js // middleware per l'autenticazione
  clearDB.js // funzione ausiliaria utilizzata durante lo sviluppo
  .env.example // esempio di file di configurazione delle
                // variabili d'ambiente
  index.js
  models // modelli dati mongoose
  package.json // file di configurazione del progetto npm
  package-lock.json
  routes // endpoint routes per le APIs
  tokenChecker.js
README.md
.gitignore // configurazione git repo
swagger
  eventTrentoAPIs.yaml // documentazione API
```

5.2 Branching strategy e organizzazione del lavoro

Malgrado i propositi di collaborazione iniziali, verso la fine del mese di settembre ho prei del gruppo è limitato al processo di brainstorming e alla realizzazione di alcuni grafici relativi allo User Flow. Tali grafici non sono presenti nella versione finale del presente documento a causa della loro modifica nel corso dello sviluppo dell'applicazione. Una descrizione dettagliata delle statistiche finali è riportata in Figura **TODO**.

```
git log --format='%aN <%aE>' | sort | uniq -c | sort -nr
```

5.3 Dependencies

Il progetto npm fa uso dei seguenti moduli:

- **Cors** per il supporto alle chiamate cross-origin
- **Express** come framework per il backend
- **Jsonwebtoken** per gestire l'autenticazione
- **Mongoose** per interfacciarsi con mongoDB

Sono inoltre riportate le seguenti dipendenze di sviluppo:

- **Dotenv** per la gestione delle variabili d'ambiente
- **TODO**

5.4 Database

La gestione dei dati è stata realizzata basandosi su un database non relazionale. In particolare, si è fatto uso del servizio offerto da Atlas MongoDB.

5.5 Testing**TODO****6 Front-End****7 Deployment**